Attorney Docket No. 1293.1995

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:

Hyun-Kwon CHUNG et al

Application No.: Unassigned Group Art Unit:

Filed: October 28, 2003 Examiner:

For: METHOD OF FOCUSING ON INPUT ITEM IN OBJECT PICTURE EMBEDDED IN MARKUP PICTURE, AND INFORMATION STORAGE MEDIUM THEREFOR

SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN APPLICATION IN ACCORDANCE WITH THE REQUIREMENTS OF 37 C.F.R. § 1.55

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, the applicant(s) submit(s) herewith a certified copy of the following foreign application:

Korean Patent Application No(s). 2002-73118

Filed: November 22, 2002

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing date(s) as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. § 119.

Respectfully submitted,

STAAS & HALSEY LLP

Date: October 28, 2003 By: _____

Michael D. Stein
Registration No. 37,240

1201 New York Ave, N.W., Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501

대 한 민 국 특 허 청
KOREAN INTELLECTUAL PROPERTY OFFICE

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto is a true copy from the records of the Korean Intellectual Property Office.

출 원 번 호 : 10-2002-0073118
Application Number

출 원 년 월 일 : 2002년 11월 22일
Date of Application NOV 22, 2002

출 원 인 : 삼성전자주식회사
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.

2003 년 08 월 08 일

특 허 청
COMMISSIONER

【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0013
【제출일자】	2002.11.22
【국제특허분류】	G06F
【발명의 명칭】	마크업 화면에 매립된 오브젝트 화면의 입력 아이템을 포커싱하는 방법 및 그 정보저장매체
【발명의 영문명칭】	Method for focusing input item on object picture embedded in markup picture and information storage medium therefor
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이영필
【대리인코드】	9-1998-000334-6
【포괄위임등록번호】	1999-009556-9
【대리인】	
【성명】	이해영
【대리인코드】	9-1999-000227-4
【포괄위임등록번호】	2000-002816-9
【발명자】	
【성명의 국문표기】	정현권
【성명의 영문표기】	CHUNG,Hyun Kwon
【주민등록번호】	721217-1042731
【우편번호】	464-800
【주소】	경기도 광주군 광주읍 탄벌리 동보아파트 104동 906호
【국적】	KR
【발명자】	
【성명의 국문표기】	정길수
【성명의 영문표기】	JUNG,Kil Soo
【주민등록번호】	750903-1917317

【우편번호】	445-970
【주소】	경기도 화성군 태안읍 병점 한신아파트 107동 707호
【국적】	KR
【발명자】	
【성명의 국문표기】	허정권
【성명의 영문표기】	HEO,Jung Kwon
【주민등록번호】	681207-1830616
【우편번호】	137-766
【주소】	서울특별시 서초구 반포2동 주공아파트 2단지 203동 504호
【국적】	KR
【발명자】	
【성명의 국문표기】	박성욱
【성명의 영문표기】	PARK,Sung Wook
【주민등록번호】	710327-1041719
【우편번호】	137-073
【주소】	서울특별시 서초구 서초3동 1595-2 센츄리오피스텔 2동 1207호
【국적】	KR
【취지】	특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인 이영필 (인) 대리인 이해영 (인)
【수수료】	
【기본출원료】	20 면 29,000 원
【가산출원료】	29 면 29,000 원
【우선권주장료】	0 건 0 원
【심사청구료】	0 항 0 원
【합계】	58,000 원
【첨부서류】	1. 요약서·명세서(도면)_1통

【요약서】

【요약】

마크업 화면에 매립된 오브젝트 화면의 입력 아이템을 포커싱하는 방법, 그 장치 및 정보저장매체가 개시된다.

본 발명에 따라 마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법에 있어서, (a) 사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 상기 오브젝트 화면을 위한 오브젝트 해석 엔진이 포커싱 이동을 위한 메시지를 상기 마크업 화면을 위한 마크업 해석 엔진으로 전달하는 단계; 및 (b) 상기 마크업 해석 엔진이 상기 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱하는 단계를 포함하는 것을 특징으로 한다. 이에 의해, 오브젝트 화면의 입력 아이템에 대해서도 마크업 화면의 입력 아이템과 같이 자유롭게 포커싱할 수 있다.

【대표도】

도 7

【명세서】

【발명의 명칭】

마크업 화면에 매립된 오브젝트 화면의 입력 아이템을 포커싱하는 방법 및 그 정보저장매체{Method for focusing input item on object picture embedded in markup picture and information storage medium therefor}

【도면의 간단한 설명】

도 1 및 2는 종래 포커싱 방식을 설명하기 위한, 인터렉티브 모드로 재생된 인터렉티브 DVD의 디스플레이 화면의 개략도,

도 3은 본 발명의 바람직한 실시예에 따른 장치의 개요도,

도 4는 본 발명에 따른 인터렉티브 컨텐츠와 프리젠테이션 엔진(1)의 소프트웨어적 구성도,

도 5는 본 발명에 따른 프리젠테이션 엔진(1)이 탑재된 장치가 포함된 재생 시스템의 구현예,

도 6은 리모트 콘트롤러(400)의 구현예,

도 7은 본 발명의 바람직한 실시예에 따른 프리젠테이션 엔진(1)의 블럭도,

도 8은 본 발명의 일 실시예에 따라 오브젝트 화면의 입력 아이템을 위한 입력 아이템 맵 정보를 설명하기 위한 참고도,

도 9는 본 발명의 일 실시예에 따라 도 2의 마크업 화면을 위한 입력 아이템 맵 정보를 설명하기 위한 참고도,

도 10은 본 발명의 다른 실시예에 따라 오브젝트 화면의 입력 아이템을 포커싱하는 방법을 설명하기 위한 참고도,

도 11은 도 10에 따라 오브젝트 화면의 입력 아이템을 포커싱하는 방법의 구현예,

도 12는 본 발명에 따라 오브젝트 화면이 매립된 마크업 화면에서 입력 아이템 간에 포커스가 이동되는 모습을 보여주는 참고도,

도 13은 본 발명의 일 실시예에 따라 오브젝트 화면이 매립된 마크업 화면에서 입력 아이템 간에 포커스가 이동되는 순서를 보여주는 참고도이다.

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<13> 본 발명은 인터렉티브 컨텐츠의 네비게이션 방식에 관한 것으로, 보다 상세하게는 마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법, 그 장치 및 정보저장매체에 관한 것이다.

<14> 인터랙티브 컨텐츠란 사용자 의사에 관계없이 일방적으로 제공되는 컨텐츠와 달리 사용자 인터페이스를 가지고 있으며 사용자 인터페이스를 통하여 사용자와의 의사소통이 가능한 쌍방향 컨텐츠를 말한다.

<15> 인터렉티브 컨텐츠의 대표적인 예로는 개인용 컴퓨터에서 재생가능한 인터렉티브 DVD를 들 수 있다. 인터렉티브 DVD는 PC를 기반으로 AV 데이터를 인터렉티브 모드로 재생할 수 있는 DVD로서, 여기에는 종래 DVD-Video 규격에 따라 AV 데이터가 담겨 있는 한편, 인터렉티브 기능을 지원하기 위한 마크업 문서가 더 기록되어

있다. 따라서, 인터렉티브 DVD에 기록된 AV 데이터는 두 가지 방법으로 디스플레이될 수 있다. 하나는 일반 DVD-Video와 동일한 방식으로 디스플레이되는 비디오 모드이고, 다른 하나는 AV 데이터가 재생되어 표시되는 AV 화면이 마크업 문서로부터 얻어진 마크업 화면에 매립되어 디스플레이되는 인터렉티브 모드이다. 예를 들어, AV 데이터가 영화 타이틀일 경우 AV 화면에는 영화가 상영되고 마크업 화면의 나머지 부분에는 영화의 대본, 줄거리, 출연배우의 사진, 등 다양한 부가 정보가 표시된다. 부가 정보는 비디오 타이틀과 동기되어 표시되기도 한다. 예를 들어, 특정 출연배우가 등장하기 시작할 때 그 배우에 대한 이력 정보가 표시되는 경우를 들 수 있다.

<16> 마크업 문서의 특정 엘리먼트는 태그를 사용하여 기록되어 있다. 특정 엘리멘트에 주어져 있는 동작은 해당 엘리먼트가 사용자에 의해 선택된 상태에서 다시 사용자의 실행명령을 받아들이는 방식에 의해 수행된다. 사용자에 의해 선택된 상태는 포커싱된 상태, 즉 "focus on" 상태라고 한다.

<17> 종래 마크업 문서의 엘리멘트를 포커싱하는 방법에는 다음과 같은 것들이 있다.

<18> 1. 마우스, 조이스틱 등과 같은 포인팅 디바이스로 해당 엘리먼트를 포커싱할 수 있다.

<19> 2. 각 엘리먼트에 미리 정해진 선택순서를 할당하고, 키보드 등과 같은 입력 디바이스를 사용하여 일 엘리먼트에서 다른 엘리먼트로 순차적으로 포커싱할 수 있다. 마크업 문서 제작자는 포커싱 순서를 "Tabbing Order"를 사용하여 정할 수 있으며, 사용자는 키보드의 tab키를 사용하여 이들 엘리먼트를 순차적으로 포커싱할 수 있다.

<20> 3. 해당 엘리먼트를 직접 포커싱할 수 있도록 엑세스 키값을 할당하고 할당된 엑세스 키값을 사용자 입력 장치로부터 입력받아 대응 엘리먼트를 포커싱한다.

<21> 그런데, 마크업 문서에 오브젝트 프로그램이 링크됨으로써, 마크업 문서로부터 얻어지는 마크업 화면에 오브젝트 프로그램으로부터 얻어지는 오브젝트 화면이 매립되어 디스플레이되는 한편 오브젝트 화면이 적어도 하나의 버튼, 링크 등 포커싱이 가능한 입력 아이템을 갖는 경우 포커싱에 문제가 생긴다.

<22> 도 1 및 2는 종래 포커싱 방식을 설명하기 위한, 인터렉티브 모드로 재생된 인터렉티브 DVD의 디스플레이 화면의 개략도이다.

<23> 도 1을 참조하면, 디스플레이 화면은 오브젝트 화면으로서 DVD-Video 화면이 매립된 마크업 화면으로 구성된다. 마크업 화면에는 포커싱할 수 있는 입력 아이템으로서 링크들, 버튼이 마련되어 있고, 오브젝트 화면에는 입력 아이템 ①, ②, ③이 존재한다.

<24> 도 2의 (a)는 마크업 화면에 마련된 링크가 포커싱되어 있음을 보여준다. 여기서, 사용자가 리모트 콘트롤러에 마련된 아래 방향키를 누르면 도 2의 (b)와 같이 포커싱이 이동된다. 사용자가 다시 좌측 방향키를 누르면 도 2의 (c)와 같이 포커싱은 좌측 엘리먼트인, DVD-Video 화면으로 이동된다. 즉, DVD-Video 화면 전체가 포커싱된다. 여기서, 다시 DVD-Video 화면 내의 입력 아이템 ①, ②, ③을 포커싱하기 위해서는 마우스 포인터를 사용해야 한다.

<25> 이처럼, 키보드나 리모트 콘트롤러 등과 같은 사용자 입력 장치를 사용한 종래 포인팅 방식에 따르면 마크업 화면에 존재하는 입력 아이템이나 오브젝트 화면에 존재하는 입력 아이템을 구분하지 않고 동일한 방식으로, 즉 종래 마크업 화면에서 입력 아이템

에 대해 포커싱하는 방식으로 포커싱할 수 없으며, 마크업 문서의 다른 엘리먼트와 같이 오브젝트 화면 전체에 대해 포커싱할 수 있을 뿐이다. 특히, 가정 내에서의 사용을 위주로 만들어지는 재생 장치에는, 사용자의 눈과 디스플레이 화면과의 거리는 마우스 포인터를 사용하여 포커싱하는 것이 사실상 불가능할 정도로 멀기 때문에, 마우스 포인터가 구비되지 않고 리모트 콘트롤러 등의 사용자 입력 장치만이 구비되므로 오브젝트 화면 내의 입력 아이템에 대한 포커싱이 더욱 문제된다.

【발명이 이루고자 하는 기술적 과제】

<26> 따라서, 본 발명의 목적은 마우스 포인터를 사용하지 않고서도 키보드, 리모트 콘트롤러 등 사용자 입력 장치를 사용하여 마크업 화면에 매립된 오브젝트 화면의 입력 아이템을 포커싱할 수 있는 방법, 그 장치 및 정보저장매체를 제공하는 것이다.

<27> 본 발명의 다른 목적은 마크업 화면에 존재하는 입력 아이템이나 오브젝트 화면에 존재하는 입력 아이템을 구분하지 않고 동일한 방식으로 포커싱할 수 있는 방법, 그 장치 및 정보저장매체를 제공하는 것이다.

【발명의 구성 및 작용】

<28> 상기 목적을 달성하기 위해, 본 발명은 마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법에 있어서, (a) 오브젝트 프로그램을 해석하여 상기 입력 아이템에 포커싱하기 위한 입력 아이템 맵 정보를 생성하는 단계; 및 (b) 사용자 입력 장치로부터의 키 입력에 응답하여 상기 입력 아이템 맵 정보를 참조하여 상기 입력 아이템 중 어느 하나에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법에 의해 달성된다.

<29> 상기 (a)단계는 XML 문서 및 자바 프로그램 중 어느 하나로 기술된 상기 오브젝트 프로그램을 해석하는 단계를 포함하는 것이 바람직하다.

<30> 상기 (a)단계는 (a1) 상기 오브젝트 프로그램으로부터 상기 입력 아이템의 입력 양식 정보, 위치 정보 및 식별자 정보를 얻는 단계; 및 (a2) 상기 입력 양식 정보, 위치 정보 및 식별자 정보를 기초로 상기 입력 아이템 맵 정보를 생성하는 단계를 포함하는 것이 바람직하다.

<31> 상기 (b)단계는 상기 사용자 입력 장치로부터 방향 키가 입력되면 상기 입력 양식 정보, 위치 정보 및 식별자 정보를 기초로 현재 포커싱된 위치에 대해 상기 방향 키가 지시하는 방향으로 가장 근접한 입력 아이템으로 포커싱을 이동하는 단계를 포함하는 것이 바람직하다.

<32> 또한, 상기 목적은 마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법에 있어서, (a) 사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 상기 마크업 화면을 위한 마크업 해석 엔진이 마크업 포커싱 이동을 위한 메시지를 상기 오브젝트 화면을 위한 오브젝트 해석 엔진으로 전달하는 단계; 및 (b) 상기 오브젝트 해석 엔진은 상기 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법에 의해서도 달성된다.

<33> 또한, 상기 목적은 마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법에 있어서, (a) 사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 상기 오브젝트 화면을 위한 오브젝트 해석 엔진이 포커싱 이동을 위한 메시지를 상기 마크업 화면을 위한 마크업 해석 엔진으로 전달하는 단계; 및 (b) 상

기 마크업 해석 엔진이 상기 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법에 의해서도 달성된다.

<34> 상기 (a)단계는 현재 포커싱된 위치 정보와 포커싱 이동을 위한 방향 정보를 담은 상기 메시지를 전달하는 단계임이 바람직하다.

<35> 상기 (b)단계는 (b1) 현재 포커싱된 입력 아이템을 기준으로 상기 방향 정보에 기초하여 선택된 방향에 대해 존재하는 다음 입력 아이템에 포커싱하는 단계를 포함하는 것이 바람직하다.,

<36> 상기 (b)단계는 (b2) 상기 방향 정보에 기초하여 현재 포커싱된 입력 아이템을 기준으로 각 입력 아이템의 거리와 지시각도를 참조하여 결정된 다음 입력 아이템에 포커싱하는 단계를 포함하는 것이 바람직하다.

<37> 한편, 본 발명의 다른 분야에 따르면 상기 목적은 마크업 언어로 작성된 마크업 문서; 및 상기 마크업 문서로부터 얻어진 마크업 화면에 매립되고 적어도 하나의 입력 아이템을 갖는 오브젝트 화면으로 디스플레이되며, 입력 아이템 맵 정보를 생성하기 위한 상기 입력 아이템의 입력 양식 정보, 위치 정보 및 식별자 정보가 담겨진 오브젝트 프로그램을 포함하는 것을 특징으로 하는 정보저장매체에 의해서도 달성된다.

<38> 상기 마크업 문서로부터 얻어진 마크업 화면에 매립되어 디스플레이되는 오디오 컨텐츠 및 이미지 컨텐츠 중 적어도 하나를 더 포함하는 것이 바람직하다.

<39> 상기 오브젝트 프로그램은 XML 문서 및 자바 프로그램과 같은 독립적 프로그램 구조를 가진 프로그램임이 바람직하다.

<40> 또한, 상기 목적은 마크업 언어로 작성된 마크업 문서; 상기 마크업 문서로부터 얻어진 마크업 화면에 매립되고 적어도 하나의 입력 아이템을 갖는 오브젝트 화면으로 디스플레이되는 오브젝트 프로그램; 및 사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 상기 오브젝트 화면을 위한 오브젝트 해석 엔진이 포커싱 이동을 위한 메시지를 상기 마크업 화면을 위한 마크업 해석 엔진으로 전달하고, 상기 마크업 해석 엔진이 상기 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱할 것을 지시하는 포커싱 전환 프로그램을 포함하는 것을 특징으로 하는 정보저장매체에 의해서도 달성된다.

<41> 상기 메시지는 현재 포커싱된 위치 정보와 포커싱 이동을 위한 방향 정보를 담고 있음이 바람직하다.

<42> 상기 포커싱 전환 프로그램은 현재 포커싱된 입력 아이템을 기준으로 상기 방향 정보에 기초하여 선택된 어느 하나의 방향에 대해 존재하는 다음 입력 아이템에 포커싱할 것을 지시하는 것이 바람직하다.

<43> 상기 포커싱 전환 프로그램은 상기 방향 정보에 기초하여 현재 포커싱된 입력 아이템을 기준으로 각 입력 아이템의 거리와 지시각도를 참조하여 결정된 다음 입력 아이템에 포커싱할 것을 지시하는 것이 바람직하다.

<44> 이하 첨부된 도면을 참조하여 본 발명의 바람직한 실시예를 상세히 설명한다.

<45> 도 3은 본 발명의 바람직한 실시예에 따른 장치의 개요도이다.

<46> 도 3을 참조하면, 장치는 프리젠테이션 엔진(1)으로 구현된다. 인터렉티브 컨텐츠는 본 발명에 따라 제작된 것으로, 사용자와 인터렉션이 가능한 인터렉티브 화면에 오브젝트 화면이 매립되어 표시되기 위한 데이터로서, 본 실시예에서는 마크업 문서와 오브젝트 프로그램을 포함한다. 마크업 문서는 인터렉티브 화면을 위한 데이터이고, 오브젝트 프로그램은 인터렉티브 화면에 매립되어 보여지기 위한 오브젝트 화면을 위한 데이터이다.

<47> 프리젠테이션 엔진(1)은 본 발명에 따른 인터렉티브 컨텐츠를 입력받아 해석하여 프리젠테이션하기 위한 장치로서, 오브젝트 프로그램을 해석하여 오브젝트 화면의 입력 아이템에 포커싱하기 위한 입력 아이템 맵 정보를 생성하고, 키보드, 리모트 콘트롤러 등의 사용자 입력 장치로부터의 키 입력에 응답하여 상기 입력 아이템 맵 정보를 참조하여 오브젝트 화면의 입력 아이템 중 어느 하나에 포커싱한다.

<48> 대안적으로, 프리젠테이션 엔진(1)은 후술하는 바와 같이 마크업 해석 엔진과 오브젝트 해석 엔진을 구비하여 사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 마크업 해석 엔진과 오브젝트 해석 엔진 간의 메시지 교환을 통해 미리 결정된 순서에 따라 입력 아이템 중 어느 하나에 포커싱한다.

<49> 도 4는 본 발명에 따른 인터렉티브 컨텐츠와 프리젠테이션 엔진(1)의 소프트웨어적 구성도이다.

<50> 도 4를 참조하면, 인터렉티브 컨텐츠는 마크업 문서 및 오브젝트 프로그램을 포함하고, 기타 컨텐츠 1, 2를 더 포함할 수 있으나 이들의 포함 여부는 선택적이다. 마크업 문서는 XML, HTML 등 마크업 언어로 작성된 문서로서 일종의 어플리케이션이다. 오브젝트 프로그램은 마크업 문서에 링크되어 마크업 화면에 매립된 애니메이션 플래시나

동영상 화면, 즉 오브젝트 화면을 보여주기 위한 프로그램으로서, 특히 본 발명에 따라 오브젝트 화면의 입력 아이템에 포커싱하기 위한 입력 아이템 맵 정보를 생성하기 위한 정보를 담고 있다. 본 실시예에서 오브젝트 프로그램은 자바(JAVA)로 코딩된다. 본 실시예에서 기타 컨텐츠 1은 사운드 데이터이고 기타 컨텐츠 2는 이미지 데이터이다.

<51> 프리젠테이션 엔진(1)은 운영체제가 탑재된 프로세서로 구현된다. 소프트웨어적 관점에서 프리젠테이션 엔진(1)에는 API(Application Program Interface)를 통해 운영체제와 통신하는 어플리케이션으로서, 오브젝트 해석 엔진과 마크업 해석 엔진이 설치되어 있다. 오브젝트 해석 엔진은 오브젝트 프로그램을 해석하고 실행하기 위한 어플리케이션이며, 마크업 해석 엔진은 마크업 문서를 해석하고 실행하기 위한 어플리케이션을 말한다. 또한, 프리젠테이션 엔진(1)에는 오브젝트 해석 엔진에 플러그-인되어 API를 통해 운영체제와 통신하는 어플리케이션인 플러그-인 1과 마크업 해석 엔진에 플러그-인되는 어플리케이션인 플러그-인 2가 설치되어 있다. 플러그-인 1은 기타 컨텐츠 1을 디코딩하기 위한 디코더이고, 플러그-인 2는 기타 컨텐츠 2를 디코딩하기 위한 디코더를 가리킨다. 플러그-인 1, 2의 구비 여부는 선택적이다.

<52> 도 5는 본 발명에 따른 프리젠테이션 엔진(1)이 탑재된 장치가 포함된 재생 시스템의 구현예이다.

<53> 도 5를 참조하면, 재생 시스템은 정보저장매체로서 디스크(100), 재생 장치(200), 본 실시예에 따른 디스플레이 장치로서 TV(300) 및 사용자 입력 장치인 리모트 콘트롤러(400)를 포함한다. 리모트 콘트롤러(400)는 사용자의 제어 명령을 수신하여 재생 장치(200)로 전달한다. 재생 장치(200)는 디스크(100)에 기록된 인터렉티브 데이터를 읽어들이기 위한 드라이브가 마련되어 있다. 드라이브에 디스크(100)가 로드되면 재생

장치(200)는 디스크에 기록된 인터렉티브 컨텐츠를 재생하여 TV(300)로 전달한다. TV(300)에는 인터렉티브 컨텐츠의 재생 화면이 표시된다. 즉, 오브젝트 프로그램으로부터 얻어진 오브젝트 화면이 매립된 마크업 화면이 표시된다. 나아가, 재생 장치(200)는 인터넷 등 네트워크에 접속하여 데이터를 송수신할 수 있다.

<54> 도 6은 리모트 콘트롤러(400)의 구현예이다.

<55> 도 6을 참조하면, 리모트 콘트롤러(400)의 전면의 상단에는 숫자 버튼과 특수 문자 버튼 그룹(43)이 배치되어 있고, 전면의 중앙에는 TV(300)의 화면에 표시되는 포커스를 위쪽으로 이동하기 위한 방향키(45), 아래쪽으로 이동하기 위한 방향키(47), 왼쪽으로 이동하기 위한 방향키(46) 및 오른쪽으로 이동하기 위한 방향키(48)가 마련되어 있으며 방향키들의 중앙에는 포인터에 의해 포인팅된 아이템을 선택하는데 사용되는 엔터키(49)가 마련되어 있다.

<56> 본 발명에 따라 사용자는 방향키들(45,46,47,48)을 사용하여 마크업 화면의 입력 아이템들 간, 오브젝트 화면의 입력 아이템들 간은 물론 마크업 화면의 입력 아이템으로부터 오브젝트 화면의 입력 아이템으로 포커스를 이동시키거나 오브젝트 화면의 입력 아이템으로부터 마크업 화면의 입력 아이템으로 포커스를 이동시킬 수 있다. 다시 말해, 마크업 화면과 오브젝트 화면을 구분하지 않고 방향키들(45,46,47,48)을 사용하여 입력 아이템들 간에 포커스를 이동시킬 수 있다.

<57> 도 7은 본 발명의 바람직한 실시예에 따른 프리젠테이션 엔진(1)의 블럭도이다.

<58> 도 7을 참조하면, 프리젠테이션 엔진(1)은 오브젝트 해석 엔진(71), 마크업 해석 엔진(72), 컨텐츠 디코더(73) 및 사용자 입력 컨트롤러(74)를 구비한다.

<59> 오브젝트 해석 엔진(71)은 오브젝트 프로그램을 해석하여 본 발명에 따른 입력 아이템에 포커싱하기 위한 입력 아이템 맵 정보를 생성하여 사용자 입력 콘트롤러(74)로 전달한다. 마크업 해석 엔진(72)은 마크업 문서를 해석하여 포커스를 받을 수 있는 엘리먼트(입력 아이템)이 존재하면 이들에 대한 본 발명에 따른 입력 아이템 맵 정보를 생성하여 사용자 입력 컨트롤러(74)로 전달한다. 사용자 입력 콘트롤러(74)는 오브젝트 해석 엔진(71)으로부터 전달받은 입력 아이템 맵 정보 및/또는 마크업 해석 엔진(72)으로부터 전달받은 입력 아이템 맵 정보를 갖게 된다. 사용자 입력 콘트롤러(74)는 리모트 콘트롤러(400)로부터의 포커싱 이동을 위한 키 입력에 응답하여 가지고 있는 입력 아이템 맵 정보를 기초로 포커싱을 이동시켜 해당 입력 아이템을 포커싱한다.

<60> 대안적으로, 오브젝트 해석 엔진(71)과 마크업 해석 엔진(72)은 리모트 콘트롤러(400)로부터의 포커싱 이동을 위한 키 입력에 응답하여 포커싱 이동을 위한 메시지를 주고 받는다. 이에, 포커싱 이동을 위한 메시지를 받은 오브젝트 해석 엔진(71) 또는 마크업 해석 엔진(72)은 메시지에 미리 결정된 순서에 따라 입력 아이템 중 어느 하나에 포커싱한다.

<61> 한편, 컨텐츠 디코더(73)는 오브젝트 해석 엔진(71)으로부터 전달받은 동영상 데이터, 이미지 데이터 및/또는 오디오 데이터를 디코딩하고 마크업 문서에 링크되어 표시되는 기타 컨텐츠를 디코딩하여 출력한다. 마크업 문서에 링크되는 기타 컨텐츠의 존재 여부는 선택적이다.

<62> 도 8은 본 발명의 일 실시예에 따라 오브젝트 화면의 입력 아이템을 위한 입력 아이템 맵 정보를 설명하기 위한 참고도이다.

<63> 도 8을 참조하면, 오브젝트 화면에는 입력 아이템으로서 이름(Name), 주소(Address), 전화번호(Telephone)을 각각 입력하기 위한 3 개의 입력폼이 마련되어 있는 한편 1 개의 OK 버튼이 마련되어 있다. 포커스는 각 입력폼들 및 OK 버튼 간을 이동할 수 있다. 이름, 주소, 전화번호를 입력할 수 있는 입력 아이템은 입력 양식이 입력폼이고, OK 버튼은 입력 양식이 "submit"이다. 이름을 입력할 수 있는 입력폼에는 식별자 정보로서 식별자 id = 1이 할당되고, 위치 정보로는 오브젝트 화면의 좌상측 꼭지점을 원점으로 했을 때 입력폼의 좌상측 꼭지점의 좌표 (x,y) = (95, 26)와, 입력폼의 좌상측 꼭지점으로부터 입력폼의 가로 길이와 세로 길이를 알려주는 정보 (cx,cy) = (84, 22)가 할당됨을 알 수 있다. 주소를 입력할 수 있는 입력폼에는 식별자 id = 2가 할당되고, 위치 정보로는 입력폼의 좌상측 꼭지점의 좌표 (x,y) = (53, 84)와, 입력폼의 가로 길이와 세로 길이를 알려주는 정보 (cx,cy) = (84, 22)가 주어진다. 전화번호를 입력할 수 있는 입력폼에는 식별자 id = 3이 할당되고, 위치 정보로는 입력폼의 좌상측 꼭지점의 좌표 (x,y) = (83, 84)와, 입력폼의 가로 길이와 세로 길이를 알려주는 정보 (cx,cy) = (84, 22)가 주어진다. 마지막으로, OK 버튼에는 식별자 id = 4가 할당되고, 위치 정보로는 버튼의 좌상측 꼭지점의 좌표 (x,y) = (56, 125)와, 입력폼의 가로 길이와 세로 길이를 알려주는 정보 (cx,cy) = (89, 26)로 구성된다.

<64> 위의 입력 아이템 맵 정보를 XML 문서로 표현하면 아래와 같다.

<65> ==========<

inputmap>

<66> <inputitemlist>

<67> ≤inputitem type="textfield" x="95" y="39" cx="84" cy="22" id="1" /> ← 이 부분을 해석(1)

<68> <inputitem type="texfield" x="95" y="53" cx="84" cy="22" id="2" />

<69> <inputitem type="textfield" x="95" y="83" cx="84" cy="22" id="3" />

<70> <inputitem type="button" x="56" y="125" cx="89" cy="26" id="4" />

<71> </itemlist>

<72> <focusinputlist>

<73> <focusitem id="1" down="2">

<74> <focusitem id="2" up="1" down="3"> ← 이부분을 해석(2)

<75> <focusitem id="3" up="2" down="4">

<76> <focusitem id="4" up="3">

<77> </focusinputlist>

<78> </inputmap>

<79> ==========

<80> 위의 XML 문서는 <itemlist>와 <focusitemlist> 두 부분으로 이루어진다. 그 중 일부를 해석하면 다음과 같다.

<81> <itemlist> 부분은 어떤 입력 아이템이 입력 포커스를 받을 수 있나를 서술하는 부분이고, <focusitemlist> 부분은 리모트 콘트롤러(400)에 마련된 방향키에 따라 어떤 입력 아이템으로 포커스를 이동해야 하는가를 서술하는 부분이다.

<82> 해석(1): 식별자 id로 "1" 값을 가지며 위치는 좌상이 (95,39)이고 폭과 높이가 (84,22)인 text field 양식의 키 입력을 받을 수 있다. 입력 양식은 "TextArea", "Button", "TextField", "List", "CheckBox" 등으로 다양하게 선정될 수 있다.

<83> 해석(2): 식별자 id로 "2" 포커스 입력이 있을 경우 상 방향의 키를 누르면 식별자 id가 "1"인 입력 아이템으로 이동하고 방향키를 누르면 식별자 id가 "2"인 입력 아이템으로 이동하게 된다. 이처럼, 입력 아이템 맵 정보를 오브젝트 프로그램인 자바 프로그램 코드 내에 기록됨으로써 이 코드가 오브젝트 해석 엔진(71)에 서 실행되어 사용자 입력 콘트롤러(74)로 전달되면 사용자는 리모콘 입력에 의해 입력 아이템들에 대한 포커스 제어를 수행할 수 있게 된다.

<84> XML 문서가 자바 프로그램에 포함되어 자바 프로그램 소스 코드로 사용되는 예는 다음과 같다.

<85> ==========

```
<86>    import java.applet.*;

<87>    public class AnimationApplet extends Applet implements Runnable {
<88>        BUTTON currentOwner;
<89>        Thread animator;

<90>        public void init()
<91>        { // applet이 로드되면 호출 됨
```

```
<92>            animator = new Thread(this);
<93>            // 입력 데이터를 받을 수 있는 입력 item들을 생성한다.
<94>            new textField(95,39,84,22,1);
<95>            new textField(95,53,84,22,2);
<96>            ...
<97>        }
<98>        public void start()
<99>        { // applet이 포함된 page를 방문하면 호출 됨
<100>           if (animator.isAive()) {
<101>              animator.resume();
<102>           }
<103>           else {
<104>              animator.start();
<105>           }
<106>       }
<107>       public void stop()
<108>       { // applet이 포함된 page를 떠나면 호출 됨
<109>          animator.suspend();
<110>       }
<111>       public void destroy()
```

```
<112>        { // 마크업 해석 엔진이 종료되면 호출됨
<113>           animator.stop();
<114>        }
<115>        public void run()
<116>        { // 스레드가 실행될 때마다 실행됨
<117>           String focus_map;

<118>           while(true) {
<119>              repaint();
<120>              Thread.sleep(100); // sleep for some time

<121>              check whether focus input is changed?
<122>              if it is changed then
<123>              {
<124>                 focus_map = get_new_focusmap(); // 새로운 Input map를 가지고 온다

<125>                 sendFocusInputMap(focus_map); // Input map을 UI controller로 전송
<126>              }
<127>           }
<128>        }
```

```
<129>         public void paint(Graphics g)
<130>         { /* Applet의 화면 출력 모양을 그려주는 함수 */
<131>            ...draw a focus indication information...
<132>            ...draws other inoformation.
<133>         }
<134>         String get_new_focusmap()
<135>         { // 새로운 입력 아이템 맵을 반환다.
<136>           // 여기서는 간단히 1개의 입력 아이템 맵을 사용하고 있으나 상황에 따라
<137>           // 입력 아이템 맵은 변경되게도 할 수 있다.
<138>            String returnmap;
<139>            returnmap = "<inputmap>"
<140>               +"<inputitemlist>"
<141>               +"<inputitem type=₩"textfield₩" x=₩"95₩" y=₩"39₩" cx=₩"84₩" cy=₩"22₩"
     id=₩"1₩" />"
<142>               +"<inputitem type=₩"texfield₩" x=₩"95₩" y=₩"53₩" cx=₩"84₩" cy=₩"22₩"
     id=₩"2₩" />"
<143>               +"<inputitem type=₩"textfield₩" x=₩"95₩" y=₩"83₩" cx=₩"84₩" cy=₩"22₩"
     id=₩"3₩" />"
<144>               +"<inputitem type=₩"button₩" x=₩"56₩" y=₩"125₩" cx=₩"89₩" cy=₩"26₩"
     id=₩"4₩" />"
```

```
<145>                 +"</itemlist>"
<146>                 +"<focusinputlist>"
<147>                 +"<focusitem id=₩"1₩" down="2">"
<148>                 +"<focusitem id=₩"2₩" up="1" down="3">"
<149>                 +"<focusitem id=₩"3₩" up="2" down="4">"
<150>                 +"<focusitem id=₩"4₩" up="3">"
<151>                 +"</focusinputlist>"
<152>                 +"</inputmap>";
<153>             return returnmap;
<154>         }
<155>     }
```

<156> ==========

<157> 위의 소스 예들은 XML DTD(Document Type Definition) 형식에 따라 다른 방식으로도 만들어질 수 있음은 물론이다.

<158> 대안적으로, XML 문서가 오브젝트 프로그램인 자바 프로그램 코드에 사용되는 것과 달리, XML 문서는 자바 프로그램 코드로 전환되어 사용될 수 있다. 그 자바 프로그램 코드의 예는 다음과 같다.

<159> ==========

```
<160> TInputMap im= new InputMap();
```

<161> TInputItem it = new TInputItem(TInputItem.TextField,95,26,84,22,-1,2,-1,-1,1);
im.add(it);

<162> TInputItem it = new TInputItem(TInputItem.TextField,95,53,84,22,1,3,-1,-1,2);
im.add(it);

<163> TInputItem it = new TInputItem(TInputItem.TextField,95,83,84,22,2,4,-1,-1,3);
im.add(it);

<164> TInputItem it = new TInputItem(TInputItem.Button,95,125,89,26,3,-1,-1,-1,4);
im.add(it);

<165> ==========

<166> 입력 아이템 맵 정보를 위한 API를 사용하여 작성된 자바 프로그램의 소스 코드의 예는 다음과 같다.

<167> ==========

<168> import java.applet.*;

<169> public class AnimationApplet extends Applet implements Runnable {

<170> BUTTON currentOwner;

<171> Thread animator;

<172> public void init()

<173> { // applet이 로드되면 호출 됨

```
<174>            animator = new Thread(this);
<175>            // 입력 데이터를 받을 수 있는 입력 아이템들을 생성한다.
<176>            new textField(95,39,84,22,1);
<177>            new textField(95,53,84,22,2);
<178>            ...
<179>         }
<180>         public void start()
<181>         { // applet이 포함된 page를 방문하면 호출됨
<182>            if (animator.isAive()) {
<183>               animator.resume();
<184>            }
<185>            else {
<186>               animator.start();
<187>            }
<188>         }
<189>         public void stop()
<190>         { // applet이 포함된 page를 떠나면 호출됨
<191>            animator.suspend();
<192>         }
<193>         public void destroy()
```

```
<194>        { // 마크업 해석 엔진이 종료되면 호출됨
<195>           animator.stop();
<196>        }
<197>        public void run()
<198>        { // 스레드가 실행될 때마다 실행됨
<199>           String focus_map;

<200>           while(true) {
<201>              repaint();
<202>              Thread.sleep(100); // sleep for some time

<203>              check whether focus input is changed?
<204>              if it is changed then
<205>              {
<206>                 // API로 입력 아이템 맵 정보를 작성하는 경우
<207>                 // 여기서는 간단히 예만을 들은 것으로 프로그램 상황에 따라
<208>                 // 입력 아이템 맵 정보가 변경되도록 할 수도 있다.
<209>                 TInputMap im = new InputMap();
<210>                 TInputItem it = new
TInputItem(TInputItem.TextField,95,26,84,22,-1,2,-1,-1,1);
```

```
<211>                   im.add(it);
<212>                   TInputItem it = new
    TInputItem(TInputItem.TextField,95,53,84,22,1,3,-1,-1,2);
<213>                   im.add(it);
<214>                   TInputItem it = new
    TInputItem(TInputItem.TextField,95,83,84,22,2,4,-1,-1,3);
<215>                   im.add(it);
<216>                   TInputItem it = new
    TInputItem(TInputItem.Button,95,125,89,26,3,-1,-1,-1,4);
<217>                   im.add(it);
<218>                   sendFocusInputMap(im); // Input map을 UI controller로 전송
<219>               }
<220>           }
<221>       }
<222>       public void paint(Graphics g)
<223>       { /* 오브젝트 화면의 출력 모양을 그려주는 함수 */
<224>           ...draw a focus indication information...
<225>           ...draws other inoformation.
<226>       }
<227>   }
```

<228> ==========

<229> 나아가, 위의 소스예들에 따른 포커스 제어 방식은 디스크를 재생하는 장치 및 캐리어 웨이브를 수신하여 재생하는 장치에도 적용시킬 수 있다.

<230> 도 9는 본 발명의 일 실시예에 따라 도 2의 마크업 화면을 위한 입력 아이템 맵 정보를 보여준다.

<231> 도 9를 참조하면, 입력 아이템 맵 정보는 입력 아이템의 입력 양식 정보, 위치 정보, 식별자 정보로 구성됨을 알 수 있다. 즉, 마크업 화면의 공룡에 대한 설명 중 입력 아이템인 공룡 이름, 예를 들어 hadrisauruses의 입력 양식(type)은 앵커(Anchor) A이고, 식별자는 id = dom: 1001이 할당되며, 그 위치 정보는 마크업 화면의 좌상측 꼭지점을 원점으로 했을 때 입력 아이템의 좌상측 꼭지점의 좌표 (x,y) = (414, 63)와, 입력폼의 좌상측 꼭지점으로부터 입력폼의 가로 길이와 세로 길이를 알려주는 정보 (cx,cy) = (140, 18)로 구성됨을 알 수 있다. 입력 아이템인 Next 버튼의 입력 양식은 "submit"이고, 식별자는 id = dom: 1010이 할당되며, 그 위치 정보는 Next 버튼의 좌상측 꼭지점의 좌표 (x,y) = (519, 439)와, Next 버튼의 좌상측 꼭지점으로부터 Next 버튼의 가로 길이와 세로 길이를 알려주는 정보 (cx,cy) = (86, 24)로 구성됨을 알 수 있다. 한편, 공룡 애니메이션이 디스플레이되는 오브젝트 화면은 마크업 화면에 대해 하나의 입력 아이템인 애니메이션 애플릿(animation applet)으로서 입력 양식은 "object"이고 식별자는 id = dom: 1011이 할당되며, 그 위치 정보는 오브젝트 화면의 좌상측 꼭지점의 좌표 (x,y) = (34, 51)와, 오브젝트 화면의 좌상측 꼭지점으로부터 오브젝트 화면의 가로 길이와 세로 길이를 알려주는 정보 (cx,cy) = (264, 282)로 구성됨을 알 수 있다.

<232> 한편, 공룡 애니메이션을 보여주는 오브젝트 화면의 입력 아이템을 위한 입력 아이템 맵 정보는 도 7의 그것과 동일한 방식으로 생성할 수 있으므로 반복되는 설명은 생략한다.

<233> 도 10은 본 발명의 다른 실시예에 따라 오브젝트 화면의 입력 아이템을 포커싱하는 방법을 설명하기 위한 참고도이다.

<234> 도 10을 참조하면, (a)에는 공룡 애니메이션을 보여주는 오브젝트 화면이 매립되어 있는 마크업 화면이 도시되어 있다. 본 실시예에 따르면 마크업 화면의 입력 아이템과 오브젝트 화면의 입력 아이템 간의 포커싱 이동은 오브젝트 해석 엔진(71)과 마크업 해석 엔진(72) 사이의 메시지 교환 방식에 따른다. 포커싱 이동을 위한 메시지 교환에 의해 오브젝트 해석 엔진(71)과 마크업 해석 엔진(72)은 포커싱 이동에 대한 제어를 넘겨받거나 넘겨준다. (a)에 도시된 화살표 방향, 즉 마크업 화면의 입력 아이템에서 오브젝트 화면의 입력 아이템으로 포커스를 이동하고자 할 때 (b)에 도시된 바와 같이 리모트 콘트롤러(400)로부터의 포커싱 이동을 위한 키 입력에 응답하여 마크업 해석 엔진(72)은 오브젝트 해석 엔진(71)으로 포커싱 이동을 위한 정보를 담은 메시지를 전달하면 오브젝트 해석 엔진(71)은 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱한다.

<235> 도 11은 도 10에 따라 오브젝트 화면의 입력 아이템을 포커싱하는 방법의 구현예를 보여준다.

<236> 도 11을 참조하면, 마크업 해석 엔진(72)은 포커스 이동을 위한 메시지로서 현재 포커싱된 위치 정보 (x,y)와 방향 정보로서 현재 포커싱된 위치에 대한 포커싱하고자 하는 위치의 방향을 알려준다. 메시지 형식은 focus change message (x,y) + direction이

다. 오브젝트 해석 엔진(71)은 메시지에 대해 승인(accept) 또는 거부(reject)를 알린다. 메시지를 승인한 경우 오브젝트 해석 엔진(71)은 현재 포커싱된 입력 아이템을 기준으로 메시지에 포함된 방향 정보에 기초하여 선택된 어느 하나에 대해 존재하는 다음 입력 아이템에 포커싱한다. 예를 들어, 사용자가 위로 이동하기 위한 방향키(45)를 눌렀다면 현재 포커싱된 입력 아이템을 기준으로 상측에 존재하는 입력 아이템 중 가장 가까이 위치하는 입력 아이템으로 포커싱을 이동한다. 상측과 좌측 또는 우측의 구분은 적절히 이루어진다.

<237> 이를 위한 포커싱 전환 프로그램의 소스 코드의 일 예는 다음과 같다.

<238> ==========

```
<239>  import java.applet.*;

<240>  public class DemandFocusApplet extends Applet {
<241>      BUTTON currentOwner;

<242>    public void paint(Graphics g)
<243>    { /* Applet의 화면 출력 모양을 그려주는 함수 */
<244>      ...draw a focus indication information...
<245>      ...draws other inoformation.
<246>    }
```

```
<247>         public boolean demandFocusOwner(int x, int y, int dir)
<248>         { /* document로부터 Focus owner가 될 수 있는 여부를 확인 받았을 때 호출되
    는 함수 */
<249>            check whether applet can get focus from parent document on dirction
    'dir' at position(x,y).
<250>            if applet can get the focus, then return (true);
<251>            eles return (false);
<252>         }

<253>         public boolean gotFocus(int x, int y, int dir)
<254>         { /* document로부터 focus를 받았을 때 호출되는 함수*/
<255>            set the button to be focused on dirction 'dir' at position(x,y).
<256>         }

<257>         public boolean keyDown(Event e, int key)
<258>         { /* 리모콘 키를 눌렀을 때 호출되는 함수 */
<259>            if applet can lose focus because user press a direction key to go out
<260>              of the focused applet, then call focus_change(key)
<261>            else
<262>              user navigates within the object boundary of the applet.
```

```
<263>       }

<264>       void focus_change(dir)
<265>       { /* focus를 눌러진 방향키에 따라 포커스 전환을 해주는 함수 */
<266>          // current focus owner is stored in currentOwner
<267>          BUTTON nextOwner;
<268>          int x, y;

<269>          x = getFocusOwnerPosition(1); // current focus position X
<270>          y = getFocusOwnerPosition(2); // current focus position Y
<271>          nextOwner = findNextFocusOwner(currentOwner,x,y,dir);

<272>          if (nextOwner == currentOwner)
<273>          {
<274>             if (notifyFocus(document,x,y,direction) == focus
<275>                accept))
<276>             {
<277>                loseFocus(currentOwner);
<278>                setFocus(document);
<279>             }
```

```
<280>                return;
<281>            }
<282>            loseFocus(currentOwner);
<283>            setFocus(nextOwner);
<284>            currentOwner = nextOwner;
<285>        }
<286>    }
<287> ============================================================================
```

<288> 도 12는 본 발명에 따라 오브젝트 화면이 매립된 마크업 화면에서 입력 아이템 간에 포커스가 이동되는 모습을 보여준다.

<289> 도 12의 (a)를 참조하면 포커스는 마크업 화면의 입력 아이템 Mongolia에 존재한다. 사용자가 리모트 콘트롤러(400)에 마련된 아래쪽으로 이동하기 위한 방향키(47)를 누르면 도 12의 (b)와 같이 포커스는 하측으로 가장 가까이 위치하는 입력 아이템 labeosaurs으로 이동된다. 다시 사용자가 좌측으로 이동하기 위한 방향키(46)를 누르면 도 12의 (c)와 같이 포커스는 좌측으로 가장 가까이 위치하는 입력 아이템 ⑥으로 이동된다. 종래 오브젝트 화면 전체에 대해 포커스가 이동되는 것과 달리, 사용자가 입장에서 보면 오브젝트 화면에 존재하는 입력 아이템들과 마크업 화면의 입력 아이템들을 전혀 구분하지 않고 포커스가 이동되는 것처럼 보인다.

<290> 도 13은 본 발명의 일 실시예에 따라 오브젝트 화면이 매립된 마크업 화면에서 입력 아이템 간에 포커스가 이동되는 순서를 보여준다.

<291> 도 13의 (a)를 참조하면, 프리젠테이션 엔진(1)(사용자 입력 컨트롤러(74))는 현재 포커싱된 입력 아이템이 좌상측에 존재할 때 사용자가 우측 방향키(49) 또는 하측 방향키(47)를 누르면 우측으로 아래 방향으로 검색해가며 다음 입력 아이템을 찾아 포커싱을 이동시킨다. 되돌아가는 방향은 별개로 정할 수 있다.

<292> 도 13의 (b)를 참조하면, 프리젠테이션 엔진(1)(사용자 입력 컨트롤러(74))는 현재 포커싱된 입력 아이템이 우하측에 존재할 때 사용자가 좌측 방향키(46) 또는 상측 방향키(45)를 누르면 좌측으로 윗 방향으로 검색해가며 다음 입력 아이템을 찾아 포커싱을 이동시킨다. 마찬가지로, 되돌아가는 방향은 별개로 정할 수 있다.

<293> 도 13의 (c)를 참조하면, 프리젠테이션 엔진(1)(사용자 입력 컨트롤러(74))는 현재 포커싱된 입력 아이템이 우상측에 존재할 때 사용자가 좌측 방향키(46) 또는 하측 방향키(47)를 누르면 아래 방향으로 각 입력 아이템의 거리와 지시각도를 참조하여 다음 입력 아이템을 찾아 포커싱을 이동시킨다. 이때 프리젠테이션 엔진(1)(사용자 입력 컨트롤러(74))는 이전에 포커싱되었던 입력 아이템에 대한 정보를 기억하고 있다가 사용자가 상측 방향키(45)를 누르면 그 순서대로 포커스를 이동시킨다.

<294> 도 13의 (d)를 참조하면, 프리젠테이션 엔진(1)(사용자 입력 컨트롤러(74))는 현재 포커싱된 입력 아이템이 우하측에 존재할 때 사용자가 상측 방향키(45)를 누르면 윗 방향으로 각 입력 아이템의 거리와 지시각도를 참조하여 다음 입력 아이템을 찾아 포커싱을 이동시킨다. 이때 프리젠테이션 엔진(1)(사용자 입력 컨트롤러(74))는 이전에 포커싱되었던 입력 아이템에 대한 정보를 기억하고 있다가 사용자가 하측 방향키(47)를 누르면 그 순서대로 포커스를 이동시킨다.

【발명의 효과】

<295> 전술한 바와 같이, 본 발명에 따르면 오브젝트 화면의 입력 아이템에 대해서도 마크업 화면의 입력 아이템과 같이 자유롭게 포커싱할 수 있다.

【특허청구범위】

【청구항 1】

마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법에 있어서,

(a) 오브젝트 프로그램을 해석하여 상기 입력 아이템에 포커싱하기 위한 입력 아이템 맵 정보를 생성하는 단계; 및

(b) 사용자 입력 장치로부터의 키 입력에 응답하여 상기 입력 아이템 맵 정보를 참조하여 상기 입력 아이템 중 어느 하나에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 2】

제1항에 있어서,

상기 (a)단계는

XML 문서 및 자바 프로그램과 같은 독립적인 프로그램 구조로 가진 상기 오브젝트 프로그램을 해석하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 3】

제1항에 있어서,

상기 (a)단계는

(a1) 상기 오브젝트 프로그램으로부터 상기 입력 아이템의 입력 양식 정보, 위치 정보 및 식별자 정보를 얻는 단계; 및

(a2) 상기 입력 양식 정보, 위치 정보 및 식별자 정보를 기초로 상기 입력 아이템 맵 정보를 생성하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 4】

제3항에 있어서,

상기 (b)단계는

상기 사용자 입력 장치로부터 방향 키가 입력되면 상기 입력 양식 정보, 위치 정보 및 식별자 정보를 기초로 현재 포커싱된 위치에 대해 상기 방향 키가 지시하는 방향으로 가장 근접한 입력 아이템으로 포커싱을 이동하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 5】

마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법에 있어서,

(a) 사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 상기 마크업 화면을 위한 마크업 해석 엔진이 마크업 포커싱 이동을 위한 메시지를 상기 오브젝트 화면을 위한 오브젝트 해석 엔진으로 전달하는 단계; 및

(b) 상기 오브젝트 해석 엔진은 상기 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 6】

마크업 화면에 매립된 오브젝트 화면의 적어도 하나의 입력 아이템을 포커싱하는 방법에 있어서,

(a) 사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 상기 오브젝트 화면을 위한 오브젝트 해석 엔진이 포커싱 이동을 위한 메시지를 상기 마크업 화면을 위한 마크업 해석 엔진으로 전달하는 단계; 및

(b) 상기 마크업 해석 엔진이 상기 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 7】

제5항 또는 제6항에 있어서,

상기 (a)단계는

현재 포커싱된 위치 정보와 포커싱 이동을 위한 방향 정보를 담은 상기 메시지를 전달하는 단계임을 특징으로 하는 방법.

【청구항 8】

제7항에 있어서,

상기 (b)단계는

(b1) 현재 포커싱된 입력 아이템을 기준으로 상기 방향 정보에 기초하여 선택된 방향에 대해 존재하는 다음 입력 아이템에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 9】

제5항 또는 제6항에 있어서,

상기 (b)단계는

(b2) 상기 방향 정보에 기초하여 현재 포커싱된 입력 아이템을 기준으로 각 입력 아이템의 거리와 지시각도를 참조하여 결정된 다음 입력 아이템에 포커싱하는 단계를 포함하는 것을 특징으로 하는 방법.

【청구항 10】

마크업 언어로 작성된 마크업 문서; 및

상기 마크업 문서로부터 얻어진 마크업 화면에 매립되고 적어도 하나의 입력 아이템을 갖는 오브젝트 화면으로 디스플레이되며, 입력 아이템 맵 정보를 생성하기 위한 상기 입력 아이템의 입력 양식 정보, 위치 정보 및 식별자 정보가 담겨진 오브젝트 프로그램을 포함하는 것을 특징으로 하는 정보저장매체.

【청구항 11】

제10항에 있어서.

상기 마크업 문서로부터 얻어진 마크업 화면에 매립되어 디스플레이되는 오디오 컨텐츠 및 이미지 컨텐츠 중 적어도 하나를 더 포함하는 것을 특징으로 하는 정보저장매체.

【청구항 12】

제10항에 있어서,

상기 오브젝트 프로그램은 XML 문서 및 자바 프로그램과 같은 독립적인 프로그램 구조를 가짐을 특징으로 하는 정보저장매체.

【청구항 13】

마크업 언어로 작성된 마크업 문서;

상기 마크업 문서로부터 얻어진 마크업 화면에 매립되고 적어도 하나의 입력 아이템을 갖는 오브젝트 화면으로 디스플레이되는 오브젝트 프로그램; 및

사용자 입력 장치로부터의 포커싱 이동을 위한 키 입력에 응답하여 상기 오브젝트 화면을 위한 오브젝트 해석 엔진이 포커싱 이동을 위한 메시지를 상기 마크업 화면을 위한 마크업 해석 엔진으로 전달하고, 상기 마크업 해석 엔진이 상기 메시지에 응답하여 미리 결정된 순서에 따라 상기 입력 아이템 중 어느 하나에 포커싱할 것을 지시하는 포커싱 전환 프로그램을 포함하는 것을 특징으로 하는 정보저장매체.

【청구항 14】

제13항에 있어서,

상기 메시지는 현재 포커싱된 위치 정보와 포커싱 이동을 위한 방향 정보를 담고 있음을 특징으로 하는 정보저장매체.

【청구항 15】

제13항 또는 제14항에 있어서,

상기 포커싱 전환 프로그램은

현재 포커싱된 입력 아이템을 기준으로 상기 방향 정보에 기초하여 선택된 어느 하나의 방향에 대해 존재하는 다음 입력 아이템에 포커싱할 것을 지시하는 것을 특징으로 하는 정보저장매체.

【청구항 16】

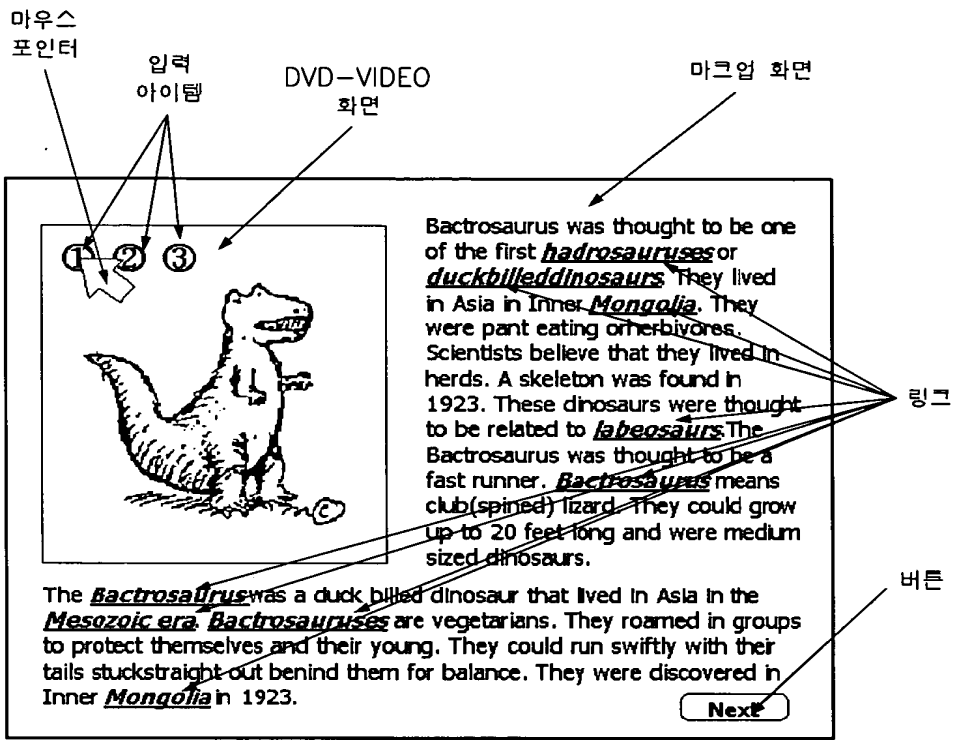
제13항 또는 제14항에 있어서,

상기 포커싱 전환 프로그램은

상기 방향 정보에 기초하여 현재 포커싱된 입력 아이템을 기준으로 각 입력 아이템의 거리와 지시각도를 참조하여 결정된 다음 입력 아이템에 포커싱할 것을 지시하는 것을 특징으로 하는 정보저장매체.

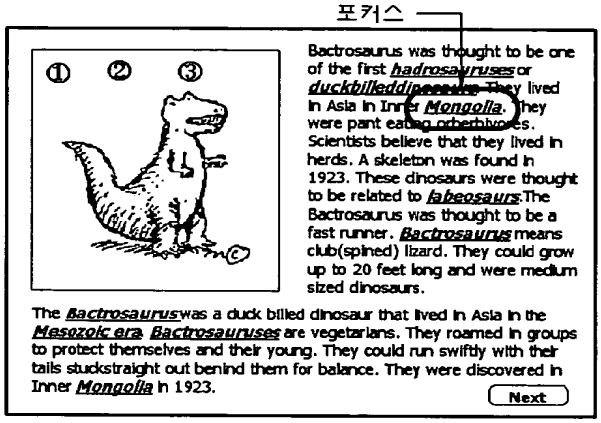
【도면】

【도 1】

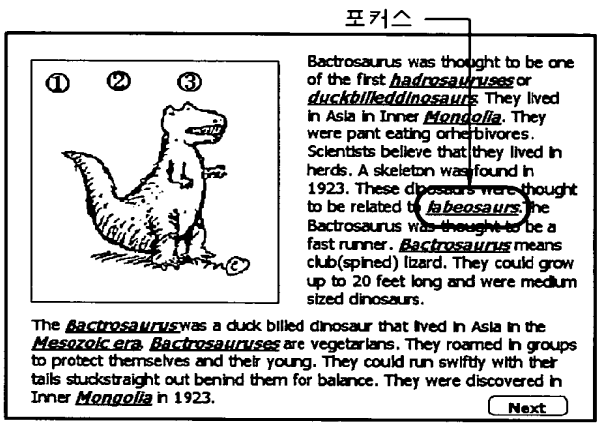

마우스
포인터
입력
아이템
DVD-VIDEO
화면
마크업 화면
링크
버튼
①
②
③
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddinosaurs. They lived in Asia in Inner Mongolia. They were pant eating orherbivores. Scientists believe that they lived in herds. A skeleton was found in 1923. These dinosaurs were thought to be related to labeosaurs. The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

【도 2】

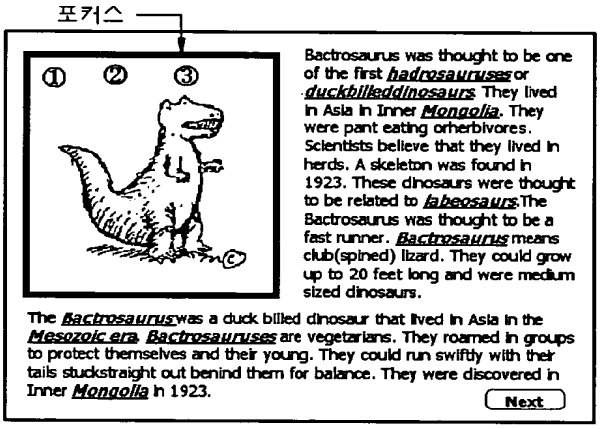
(a)


포커스
①
②
③
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddinosaurs. They lived in Asia in Inner Mongolia. They were pant eating orherbivores. Scientists believe that they lived in herds. A skeleton was found in 1923. These dinosaurs were thought to be related to Iabeosaurs.The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

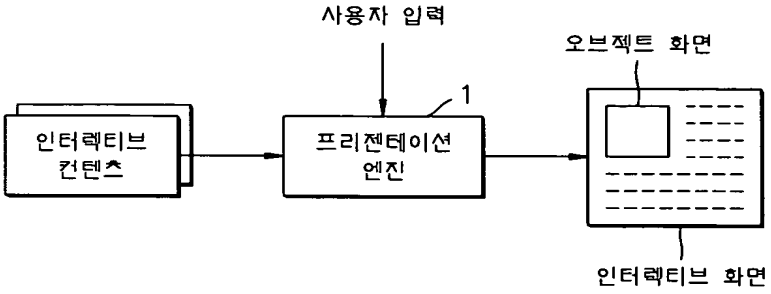
(b)


포커스
①
②
③
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddinosaurs. They lived in Asia in Inner Mongolia. They were pant eating orherbivores. Scientists believe that they lived in herds. A skeleton was found in 1923. These dinosaurs were thought to be related to Iabeosaurs.The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

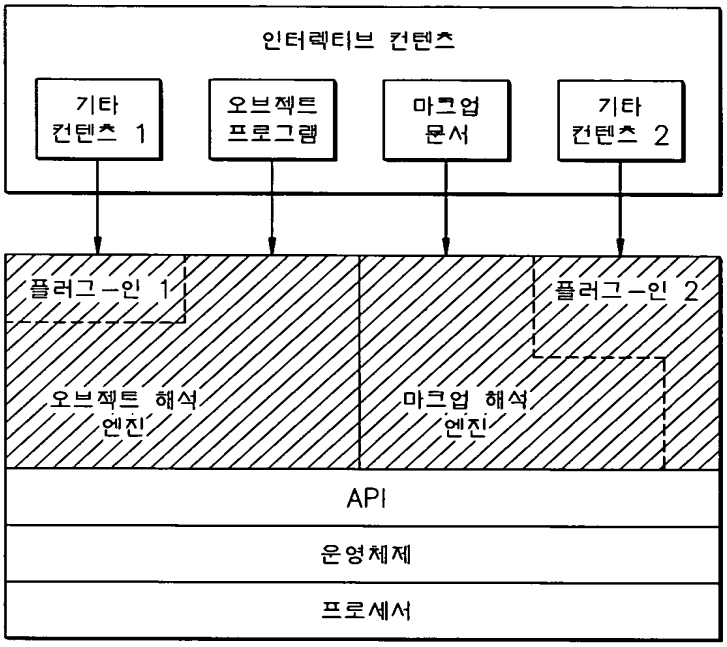
(c)


포커스
①
②
③
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddinosaurs. They lived in Asia in Inner Mongolia. They were pant eating orherbivores. Scientists believe that they lived in herds. A skeleton was found in 1923. These dinosaurs were thought to be related to Iabeosaurs.The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

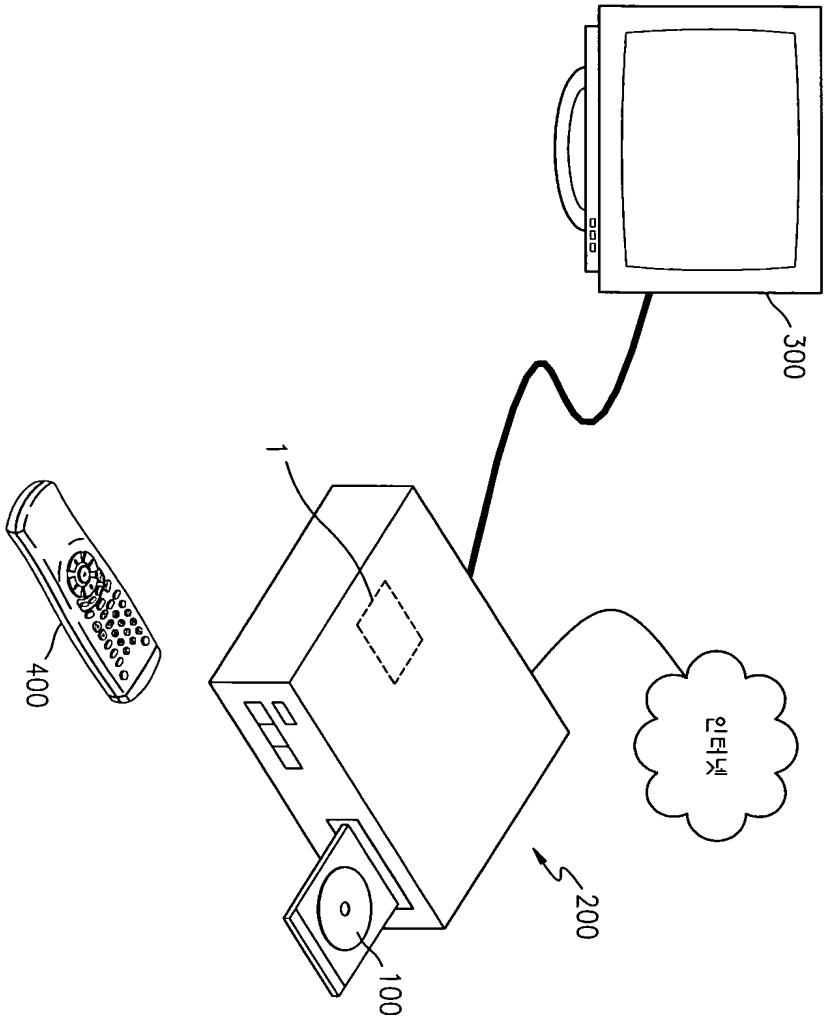
【도 3】


사용자 입력
오브젝트 화면
1
인터렉티브
컨텐츠
프리젠테이션
엔진
인터렉티브 화면

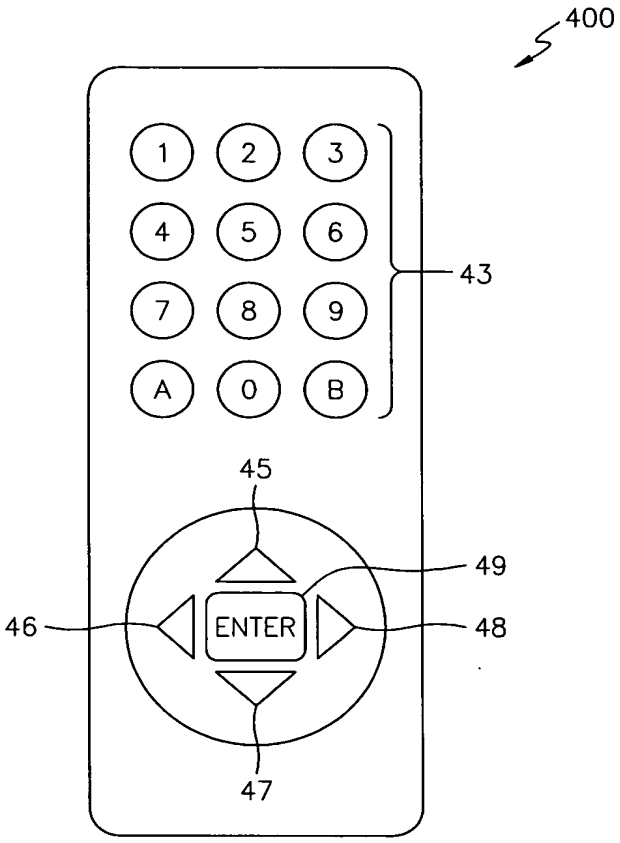
【도 4】


인터렉티브 컨텐츠
기타
컨텐츠 1
오브젝트
프로그램
마크업
문서
기타
컨텐츠 2
플러그-인 1
플러그-인 2
오브젝트 해석
엔진
마크업 해석
엔진
API
운영체제
프로세서

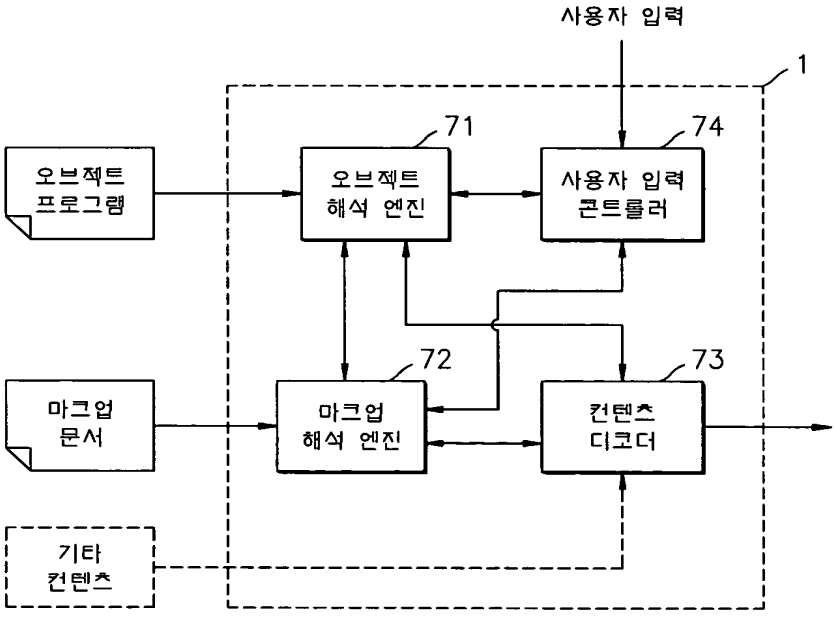
【도 5】



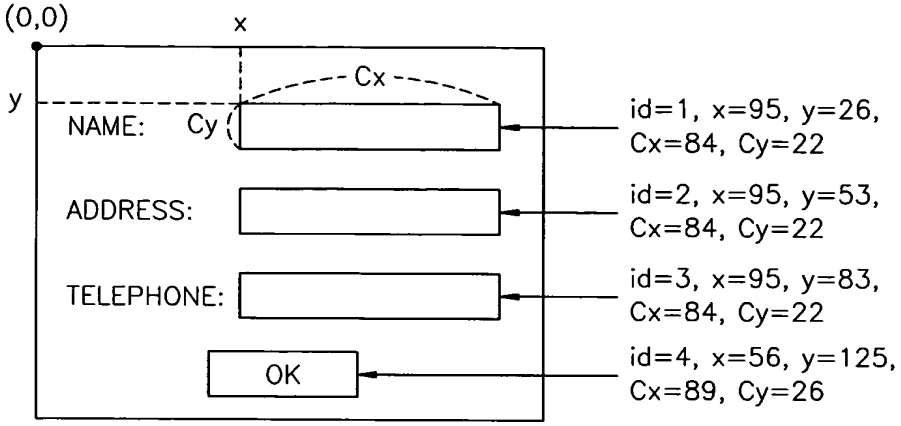
【도 6】


400
1
2
3
4
5
6
7
8
9
A
0
B
43
45
46
ENTER
48
49
47

【도 7】


사용자 입력
1
71
74
오브젝트 프로그램
오브젝트 해석 엔진
사용자 입력 콘트롤러
72
73
마크업 문서
마크업 해석 엔진
컨텐츠 디코더
기타 컨텐츠

【도 8】

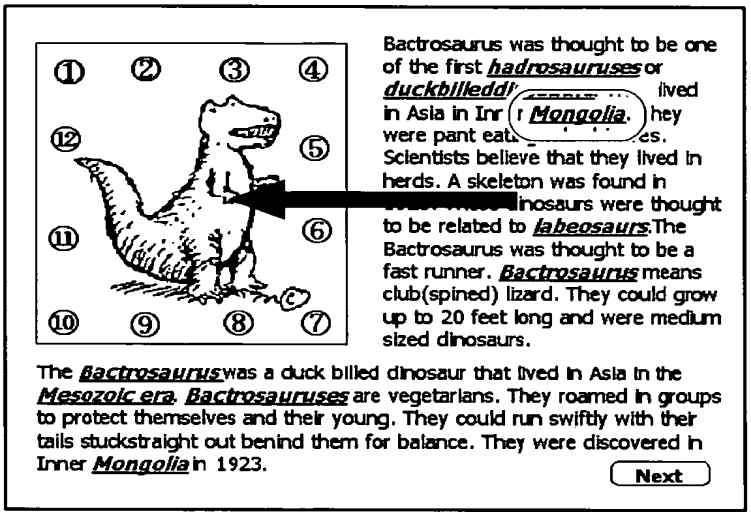

(0,0)
x
y
Cx
Cy
NAME:
ADDRESS:
TELEPHONE:
OK
id=1, x=95, y=26,
Cx=84, Cy=22
id=2, x=95, y=53,
Cx=84, Cy=22
id=3, x=95, y=83,
Cx=84, Cy=22
id=4, x=56, y=125,
Cx=89, Cy=26

【도 9】

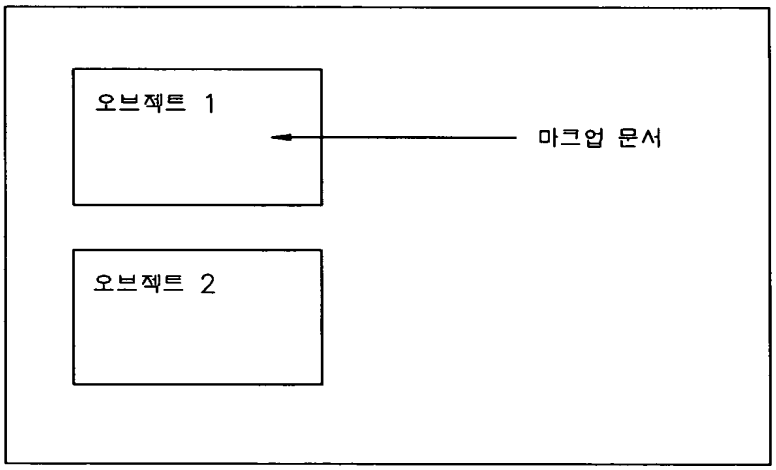
입력 아이템	type	x	y	cx	cy	id
hadrosauruses	A	414	63	40		dom:1001
duckbilleddinosaurs	A	328	84	58	18	dom:1002
Mongolia	A	451	107	85	18	dom:1003
labeosaurus	A	453	212	11	18	dom:1004
Bactrosaurus	A	452	254	23	18	dom:1005
Bactrosaurus	A	69	347	23	18	dom:1006
Mesozoic era	A	36	372	18	18	dom:1007
Bactrosauruses	A	161	372	41	18	dom:1008
Mongolia	A	81	433	85	18	dom:1009
[Next]	submit	519	439	86	24	dom:1010
에니메이션 에플릿	object	34	5	2	282	dom:1011

【도 10】

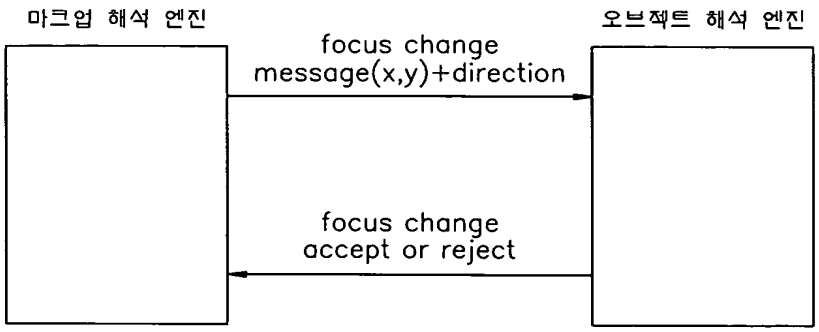
(a)


①
②
③
④
⑤
⑥
⑦
⑧
⑨
⑩
⑪
⑫
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddi lived in Asia in Inr r Mongolia. hey were pant eat es. Scientists believe that they lived in herds. A skeleton was found in inosaurs were thought to be related to labeosaurs.The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

(b)

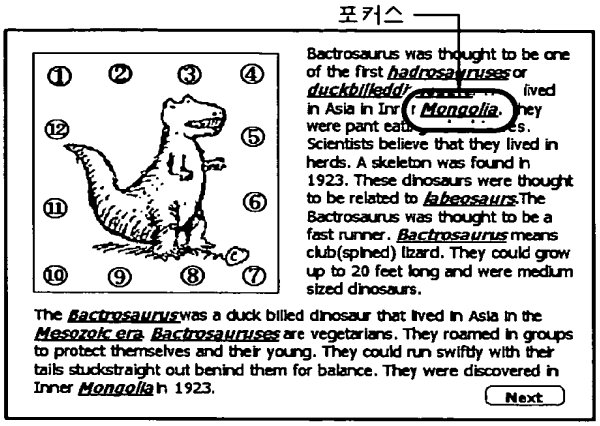

오브젝트 1
마크업 문서
오브젝트 2

【도 11】

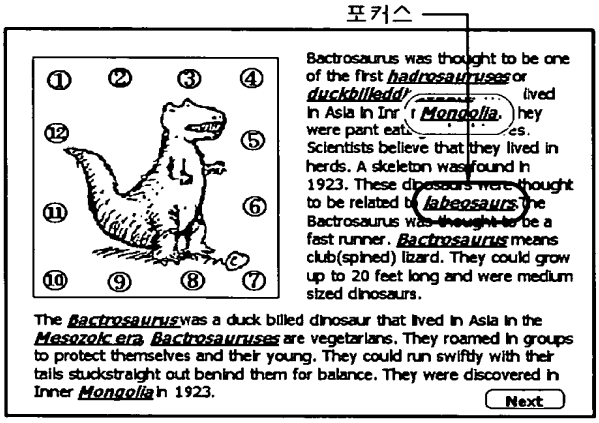

마크업 해석 엔진
오브젝트 해석 엔진
focus change
message(x,y)+direction
focus change
accept or reject

【도 12】

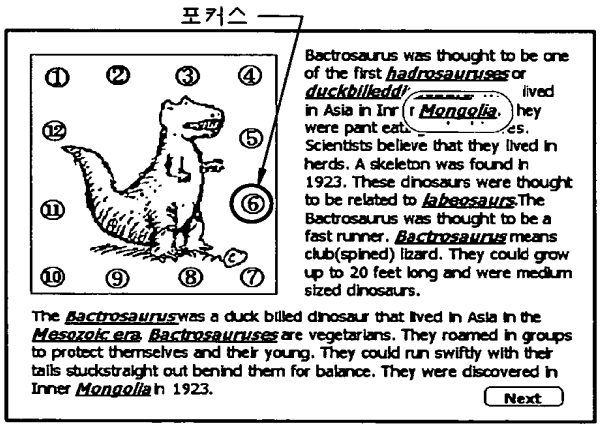
(a)


포커스
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddi lived in Asia in Inr r Mongolia. hey were pant eat es. Scientists believe that they lived in herds. A skeleton was found in 1923. These dinosaurs were thought to be related to Iabeosaurs.The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

(b)

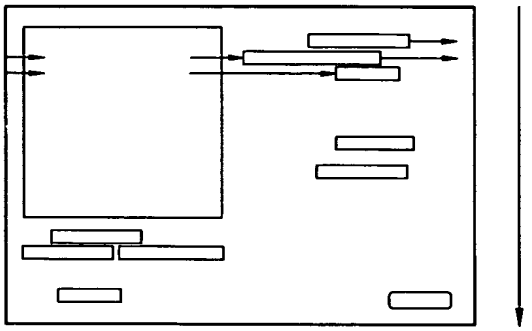

포커스
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddi lived in Asia in Inr r Mongolia. hey were pant eat es. Scientists believe that they lived in herds. A skeleton was found in 1923. These dinosaurs were thought to be related to Iabeosaurs.The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

(c)

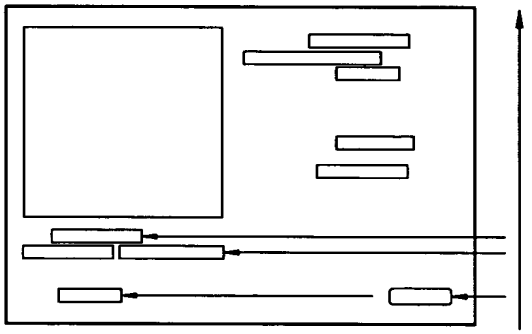

포커스
Bactrosaurus was thought to be one of the first hadrosauruses or duckbilleddi lived in Asia in Inr r Mongolia. hey were pant eat es. Scientists believe that they lived in herds. A skeleton was found in 1923. These dinosaurs were thought to be related to Iabeosaurs.The Bactrosaurus was thought to be a fast runner. Bactrosaurus means club(spined) lizard. They could grow up to 20 feet long and were medium sized dinosaurs.
The Bactrosaurus was a duck billed dinosaur that lived in Asia in the Mesozoic era. Bactrosauruses are vegetarians. They roamed in groups to protect themselves and their young. They could run swiftly with their tails stuckstraight out benind them for balance. They were discovered in Inner Mongolia in 1923.
Next

【도 13】

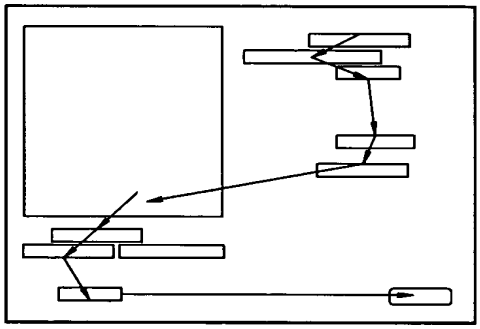
(a)



(b)



(c)



(d)